並列テキスト生成による低遅延ゲーム音声実況システム

川松亮太 [1,2]　齋藤佑樹 [1]　高道慎之介 [3,1]　ニュービッググラム [4]
須藤克仁 [5]　高村大也 [2]　石垣達也 [2]
[1] 東京大学　[2] 産総研　[3] 慶應義塾大学　[4]CMU　[5] 奈良女子大学
kawamatsu-ryota@g.ecc.u-tokyo.ac.jp

概要

本研究では、ゲーム映像を逐次入力し実況音声を出力するリアルタイム音声実況生成タスクを扱う。従来、テキスト生成と音声合成を直列に行う手法が用いられていたが、各段階での推論時間の蓄積により実況音声が出力されるまでの待ち時間が発生し、長い沈黙を含む不自然な実況出力が問題であった。そこで、本稿ではテキスト生成を並列化することで待ち時間を軽減する音声実況システムを提案する。ゲーム実況コーパスを用いた評価実験より、提案システムによる自動生成実況はベースラインと比較して、不自然な実況の原因となる発話間の平均沈黙時間を 9.6 秒から 0.3 秒にまで軽減した。また、発話と沈黙の繰り返し時間のパターンについても、mean Intersection over Union (mIoU) によるプロによる実況との類似度評価において 40 %以上の性能向上を確認した。さらに、人手評価により提案法は実況の発話リズムの自然さを向上させることが示された。

1　はじめに

ゲーム実況は、進行中のゲームに対して実況者が状況や展開を言語化し、視聴者に向けて発信する行為である。実況者による状況描写や補足説明を通じて、視聴者がゲーム内容を理解しやすくなるほか、実況者の巧みな話術によりゲーム実況そのものを娯楽として楽しむことができる [1, 2]。ゲーム実況は高度な話術や分野知識が必要であり、誰しもが容易に行えるものではない。そのため、自然言語処理技術を用いて、ゲーム映像から実況を自動生成する研究が行われている [3]。

従来、実況生成に関する研究は、テキスト生成と音声合成の分野で独立して発展している。テキスト生成の分野では、サッカーやゲーム等を対象に、刻一刻と変化する状況を正確に言語化することに主眼が置かれてきた [4, 5, 6]。一方、音声合成においては、例えば、Iura ら [7] は戦況の盛り上がりを反映した抑揚制御について提案している。これらを統合する研究は少なく、例えば、Ishigaki ら [8] は BART [9] によるテキスト生成と音声合成を接続した単純な直列構成を提案している。

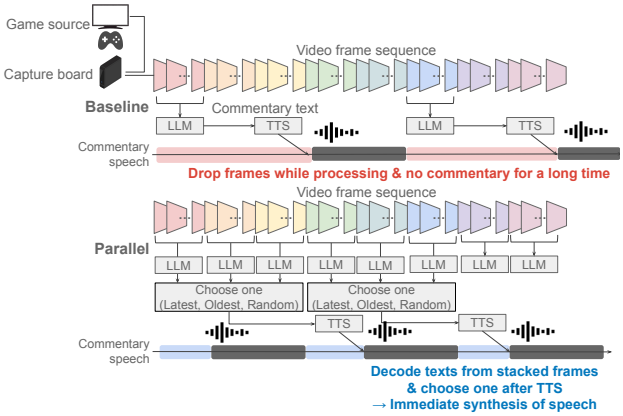

図 1: 音声実況生成システムの構成。従来の実況システム（図中 Baseline）は、映像取得、テキスト生成、音声合成を順次実行する。対して提案システム（図中 Parallel）は、テキスト生成を並列化し、生成された実況文をテキストバッファに蓄積する。

このような直列構成では各モジュールの推論時間が蓄積し、音声実況の出力までの待ち時間が問題となる。この待ち時間は長い沈黙時間を含む実況を生み出し、不自然な実況の要因の一つとなる。

本稿では待ち時間を軽減するため、並列処理可能なテキスト生成機構を持つリアルタイム音声実況システムを提案する。提案システムでは待ち時間低減のためにテキスト生成の並列化を行う。複数のテキスト生成処理を非同期に動作させることで、前の発話が終わるのを待たずに次の発話生成を開始し、待ち時間を軽減する。

SMASH コーパス [10] を用いた評価実験より、並列生成を行わないベースラインは許容可能な音声遅延（約 3.71 秒） [11] を大きく超えることを確認し

た。一方で、提案手法は不自然な実況の原因となる発話間の平均沈黙時間を 9.6 秒から 0.3 秒にまで軽減することが分かった。発話と沈黙の繰り返し時間のパターンについても、mean Intersection over Union (mIoU) によるプロ実況音声との類似度評価において 40 %以上の性能向上を確認し、提案システムがプロ実況者により近い発話パターンであることが確かめられた。さらに、120 人の主観評価においても、提案システムによる実況は発話時間と沈黙時間の繰り返しパターンがより自然であると評価された。

2 リアルタイム音声実況生成

従来の実況生成システムでは、映像取得、テキスト生成、音声合成を順次実行する直列的な構成を採用している [8, 12]。

リアルタイム音声実況生成システムの構成を図 1 に示す。このシステムでは、ユーザがゲームをプレイし、プレイ映像がディスプレイに送出される。まず、プレイ中のゲーム映像はキャプチャボードを介してシステムに取り込まれ、フレームバッファに蓄積される。具体的には、ゲーム映像はフレーム単位で時系列に取得され、連続する N 枚のフレーム $\{f_i, \ldots, f_{i+N-1}\}$ を 1 つの単位としてフレーム列 F_k が順次形成される。ここで、f_i は時刻 i に取得されたフレームを表し、F_k は k 番目に形成されるフレーム列（ブロック）である。

次に、形成されたフレーム列 F_k を大規模言語モデル (LLM) に入力し、ゲーム状況を反映した実況テキストを生成する。この際、時系列的に生成される複数のフレーム列のうち、入力時点でもっとも新しいフレーム列を用いる [12]。モデルにはマルチモーダル LLM を採用する。テキストを生成する際には、マルチモーダル LLM に対してゲームプレイ映像を与え、「ゲームの状況を説明する実況テキストを生成してください。話すことがなければ沈黙してください。」という単純なプロンプトを入力する。自動生成した発話履歴は将来的な生成や発話選択のためにテキストバッファ内に保持しておく。

最後に、生成されたテキストを音声合成モジュールに入力し、実況音声を再生する。音声合成モデルには、既存のテキスト読み上げソフトウェアや、Iura ら [7] により構築されたゲーム実況解説に焦点を当てたモデルなどが用いられる。

しかし、この直列的なシステム構成では、ある発話の音声再生が完全に終了してから次のフレーム取得およびテキスト生成を開始する。そのため、後続する発話のテキスト生成・音声合成が完了するまでの間、必然的に長い沈黙が発生してしまう。これが不自然な実況を生成する一要因となっていた。

3 提案手法

前節で述べた直列構成にテキスト生成の並列化を加えた新たな音声実況生成システムを提案する。

あるフレーム列 F_k の処理中であっても、次なるフレーム列 F_{k+1} が形成され次第、即座にテキスト生成を並列して開始する。これにより、テキストバッファに未読の実況文が複数格納された状態を維持する。前の発話音声の再生が完了し次第、次の発話をテキストバッファから選択し音声合成することで、テキスト生成処理の完了を待つこと無く途切れのない実況を実現する。発話選択は現在発話中の実況文が発話を開始してから終了するまでの間にテキストバッファに蓄積された実況文を対象に行う。具体的には以下の 3 つの発話選択手法を提案する：

- **Parallel Latest**: 最も新しいフレーム列に対応する実況文を採用する。
- **Parallel Oldest**: 最も古いフレーム列に対応する実況文を採用する。
- **Parallel Random**: 任意のフレーム列に対応する実況文をランダムに採用する。

4 実験

4.1 実装

ゲーム映像再生用のディスプレイと実況生成用コンピュータの 2 台をキャプチャーボード (Elgato HD60 X[1]) を介して接続し、ユーザが実際にプレイしながら実況が自動生成される状況を模擬した。映像は 25 fps のフレームレートで取得し、言語生成モデルには 32 フレーム分を base64 によりエンコードしたトークン列として与えた。音声合成モデルのパラメータは [7] に準拠した。実況テキスト生成には GPT-4.1-mini を用いた。提案手法では主に沈黙時間を制御するための機構を提案しているが、発話時間を制御するためには LLM の推論時に生成される最大トークン数を制限する max new tokens パラメータ設定が重要である。そこで、本研究では max new tokens を $\{20, 40, 60, 80, 100\}$ の 5 段階に設定しその

1） https://www.elgato.com/jp/ja/p/game-capture-hd60-x

影響を比較する。

テキスト生成と音声合成にかかる処理時間を 0 にすることは原理的に不可能である。そこで、少なからず遅延が発生するという前提のもと、映像の送出をあえて遅延させユーザに提示することで、自動生成実況音声と映像を同期させる機構もすべての比較手法に実装する。具体的には、映像をバッファリングし、最初のフレーム列に対応する実況音声の再生開始時刻に合わせて、対応する映像の再生を開始することで同期を図る。つまり、映像を最初の発話の生成と音声合成にかかる時間だけ遅延させる。

4.2 データセット

実験には、大乱闘スマッシュブラザーズ SPECIAL を対象としたデータセットを用いた。本タイトルはゲームの進行ペースが速く、遅延の影響がより顕著である。具体的には、映像データとして SMASH コーパス [10] に収録されているゲームプレイ映像を使用した。一方で、実況音声については同コーパスのものではなく、当該映像に対してプロの実況者が行った実況音声を独自に収集したものを用いた。データセット全体から 8 本のゲーム映像を評価対象として無作為に選択した。

4.3 比較手法

以下の 5 つの手法を比較する。ただし、4.1 節で述べたようにいずれの手法も映像放出を遅延させ、できる限り実況と映像が同期するよう実装した：

- **Baseline After-Audio**: 音声再生完了を待機してから次の実況生成を開始する逐次的手法。既存の実装手法である [8]。
- **Baseline After-Text**: テキスト生成終了時に音声再生を待たずに次フレームの実況生成を開始する半逐次的手法。
- **Parallel Latest**: 音声再生終了時点でテキストバッファ内の最新の発話を選択する並列手法。
- **Parallel Oldest**: 音声再生終了時点でテキストバッファ内の最も古い発話を選択する並列手法。
- **Parallel Random**: 音声再生終了時点でテキストバッファ内の任意の発話を選択する並列手法。

4.4 評価指標

自動評価および人間の評価者による採点により、1) 発話と沈黙の繰り返し傾向の自然性に加え、2) 発話内容の妥当性を評価する。提案手法が特に発話と沈黙の繰り返し傾向の自然性の向上に寄与することを期待している。

自動評価：発話時間や沈黙時間、その繰り返し傾向がプロの実況者に類似していれば、より自然な実況であると仮定する。この考えのもと、発話・沈黙時間の累積および平均、実況文長を計測し、プロの実況者の値と比較する。加えて、発話状態を 1、沈黙を 0 とし、サンプリング間隔が 1 秒の時系列ベクトルとして実況の発話-沈黙パターンを表現する。このベクトルを用いて、人間のプロ実況との mean Intersection over Union (mIoU) を算出する。すなわち、プロ実況が発話している時刻に自動生成実況も発話し、プロ実況が沈黙する時刻に自動生成実況も沈黙していれば高いスコアを得る。これにより、発話と沈黙の繰り返し規則の一致度を定量化する。

発話内容に関しては、正解実況文に対する ROUGE スコアによる自動評価を行った。なお、ROUGE の算出にあたっては、時系列のズレを考慮し、10 秒単位のセグメントごとに評価を実施した [2)]。

人手評価：評価者 120 名に対し、5 つの手法を 5 通りの max new tokens パラメータを設定した計 25 通りの音声実況付き動画を無作為に提示した。なお、評価者の負担を考慮し、約 3 分の元映像から共通のタイミングで 30 秒切り出したクリップを評価事例とした。評価対象のシーンは計 20 種類用意した。各評価事例を以下の 3 つの観点から採点者に 5 段階で評価させた：

- Q1. 発話リズムの自然さ：実況の内容ではなく、喋りのリズムや間の取り方
- Q2. 映像との一致度：実況の内容が映像（ゲームの状況）と正しく合っているか
- Q3. 実況としての全体的な質：リズム、内容、雰囲気などを含めた実況解説としての総合評価

評価者はクラウドソーシング (Lancers[3)]) で募集した。一般向けのゲーム実況を想定しているため、評

2) BERTScore による意味空間上での距離による評価は多くのテキスト生成研究において用いられているが、予備実験において本研究では手法間の差を観測しなかった。発話はすべて同一分野を対象にしたものであり、意味ベクトル空間上で近い場所に分布したためと考えている。

3) https://www.lancers.jp/

表 1: 各手法における沈黙時間の統計量および mIoU。ここでは max new tokens を 20 に固定した。括弧内の数値は標準偏差を表す。

手法	累積	平均	mIoU
Human	59.7 (± 11.3)	1.7 (± 0.4)	–
After-Audio	134.0 (± 5.2)	9.5 (± 0.9)	0.01 (± 0.06)
After-Text	125.5 (± 5.8)	6.8 (± 0.9)	0.10 (± 0.08)
Latest	24.7 (± 6.6)	0.4 (± 0.1)	0.59 (± 0.04)
Oldest	18.9 (± 3.1)	0.3 (± 0.1)	0.60 (± 0.04)
Random	19.1 (± 3.6)	0.3 (± 0.1)	0.60 (± 0.03)

表 2: max new tokens を変化させたときの発話時間、および発話長の統計量。ここでは実況文生成手法に Latest を用いた。括弧内の数値は標準偏差を表す。

max new tokens	発話時間の平均	発話長
Human	2.8 (± 0.5)	25.9 (± 4.1)
20	2.3 (± 0.1)	16.8 (± 0.4)
40	2.9 (± 0.1)	22.7 (± 0.6)
60	3.4 (± 0.1)	28.2 (± 0.8)
80	3.8 (± 0.1)	31.8 (± 1.1)
100	4.4 (± 0.1)	36.3 (± 1.3)

価者の属性は指定せず幅広い属性から募った。

5 結果

表 1 に各手法における沈黙時間、発話量に関する統計量および mIoU を示す。ベースライン群（After-Audio, After-Text）は処理の待ち時間の影響により Human の約 2 倍の累積沈黙時間を記録し、不自然な間が生じていることが示された。対して提案手法群（Latest, Oldest, Random）は、並列化により沈黙時間を大幅に短縮し、mIoU も 0.60 程度とプロの実況に近い発話リズムを実現した。なお、提案手法内での選択方策による性能差は軽微であった。

発話時間について自動生成実況と人間を比較するため、Latest において max new tokens を変化させた際の発話時間および発話長（文字数）を表 2 に示す。表より、max new tokens を 40 に設定した場合、発話時間の平均が Human に最も近く、発話長は 60 に設定した場合が最も Human に近い。このことから、人間の発話時間に近い実況を生成するには、本パラメータの調整が重要であることが分かる。

また、内容の一致度を示す ROUGE スコア（図 2）においても、提案手法群はベースライン群を大幅に上回った。ただし、80〜100 トークンの設定では Latest 手法にスコア劣化が見られ、大きすぎる max new tokens パラメータの値は実況の品質を劣化させる。よって、適切な max new tokens パラメータも重要であることがここでも示された。

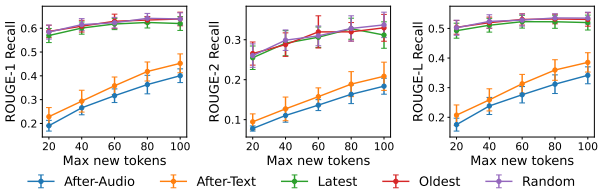

図 2: 各手法における max new tokens の変化に伴う ROUGE スコアの推移。左から順に ROUGE-1 Recall、ROUGE-2 Recall、ROUGE-L Recall を示す。エラーバーは標準偏差を表す。

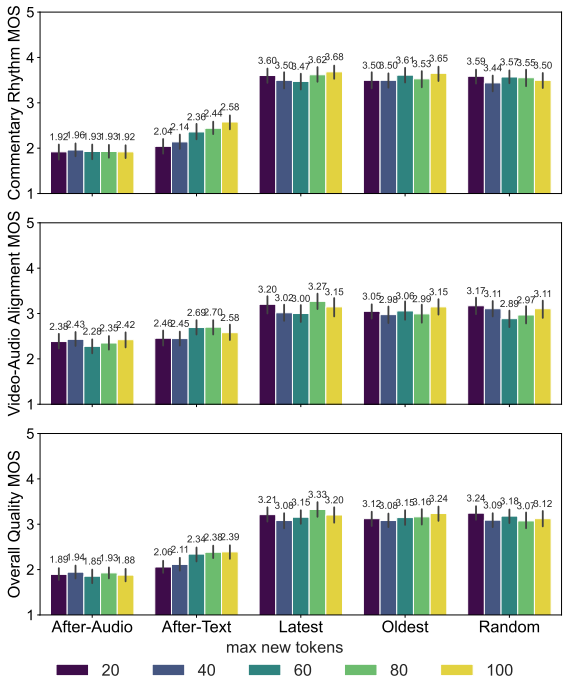

図 3: 各手法における max new tokens (20, 40, 60, 80, 100) ごとの主観評価結果。上から Q1 から Q3 の評価結果を示す。エラーバーは 95 %信頼区間を表す。

図 3 に人手評価の結果を示す。Tukey の HSD 法（$\alpha = 0.05$）による多重比較の結果、Q1、Q2、Q3 の全項目において、提案手法群はベースライン群よりも有意に高いスコアを示した。これは、発話間の不自然な沈黙を削減したことが、実況テンポの改善のみならず、視聴体験の全体的な向上に寄与したことを示唆している。

6 おわりに

本稿では並列テキスト生成により音声実況生成システムの待ち時間を軽減する手法を提案した。処理の待ちによる不自然な沈黙時間を軽減したことが自動評価および人手評価により確かめられた。

謝辞

本研究には、内閣府が実施する「研究開発成果の社会実装への橋渡しプログラム（BRIDGE）/AI ×ロボット・サービス分野の実践的グローバル研究」により得られた成果が含まれています

参考文献

[1] Anton Behrens and Sebastian Uhrich. You’ll never want to watch alone: the effect of displaying in-stadium social atmospherics on media consumers’ responses to new sport leagues across different types of media. **European Sport Management Quarterly**, 2022.

[2] Jaime A. Teixeira da Silva and Nicolas Scelles. A vision for the formal documentation and digitalization of sports commentators’ commentaries. **International Journal of Sport Communication**, 2025.

[3] Qirui Zheng, Xingbo Wang, Keyuan Cheng, Muhammad Asif Ali, Yunlong Lu, and Wenxin Li. From multimodal perception to strategic reasoning: A survey on ai-generated game commentary, 2025.

[4] Yasufumi Taniguchi, Yukun Feng, Hiroya Takamura, and Manabu Okumura. Generating live soccer-match commentary from play data. In **AAAI**, 2019.

[5] Joya Chen, Ziyun Zeng, Yiqi Lin, Wei Li, Zejun Ma, and Mike Zheng Shou. Livecc: Learning video llm with streaming speech transcription at scale. In **Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)**, 2025.

[6] Yuichiro Mori, Chikara Tanaka, Aru Maekawa, Satoshi Kosugi, Tatsuya Ishigaki, Kotaro Funakoshi, Hiroya Takamura, and Manabu Okumura. Live football commentary system providing background information. In **Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 3: System Demonstrations)**, 2025.

[7] Kota Iura, Yuki Saito, Shinnosuke Takamichi, Graham Neubig, Katsuhito Sudoh, Hiroshi Saruwatari, Hiroya Takamura, and Tatsuya Ishigaki. Excitement-inducing commentary text-to-speech system for fighting game video scenes. **IEEE Access**, 2025.

[8] Tatsuya Ishigaki, Goran Topić, Yumi Hamazono, Ichiro Kobayashi, Yusuke Miyao, and Hiroya Takamura. Audio commentary system for real-time racing game play. In **Proceedings of the 16th International Natural Language Generation Conference: System Demonstrations**, 2023.

[9] Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Veselin Stoyanov, and Luke Zettlemoyer. BART: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. In **Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics**, 2020.

[10] Yuki Saito, Shinnosuke Takamichi, and Hiroshi Saruwatari. SMASH corpus: A spontaneous speech corpus recording third-person audio commentaries on gameplay. In **Proceedings of the Twelfth Language Resources and Evaluation Conference**, 2020.

[11] Ryosuke Matsushita, Ryosuke Sakai, Koki Fukuda, Shinnosuke Takamichi, Kota Iura, Yuki Saito, Graham Neubig, Katsuhito Sudoh, Hiroya Takamura, and Tatsuya Ishigaki. Measuring time delay tolerance in third-person live commentary for super smash bros. ultimate. In **2025 IEEE Conference on Games (CoG)**, 2025.

[12] Tatsuya Ishigaki, Goran Topic, Yumi Hamazono, Hiroshi Noji, Ichiro Kobayashi, Yusuke Miyao, and Hiroya Takamura. Generating racing game commentary from vision, language, and structured data. In **INLG**, 2021.